2018-08-13

Vocable Code: Lecture-performance in six parts

Cox, Geoffrey

VOCABLE CODE (13082018)

A lecture-performance in six parts
Winnie Soon & Geoff Cox

Vocable Code is both a work of 'software art' (software as artwork, not software to make an artwork) and a 'codework' (where the source code and critical writing operate together) produced to embody "queer code". Collective statements and voices complete the phrase 'Queer is…' and together make a computational and poetic composition for two screens: on one of these, texts and voices are repeated and disrupted by mathematical chaos, together exploring the performativity of code and language; on the other, is a mix of a computer programming syntax and human language. In this sense queer code can be understood as both an object and subject of study that intervenes in the world's 'becoming' and how material bodies are produced via human and nonhuman practices.
The purpose is to exemplify the speech-like qualities of a computer program, and to explore the constant regeneration and re-running of code as a way to rethink computational logic from a posthuman position.

Performer #1:
· Omit { } , ; ( ) ' '
· Pronounce [ ] . =
· Pause deliberately for non-pronounced symbols

```
let whatisQueer;
let queerRights = [];
let speak;
let queers = [];
let voices = [];
function preload() {
  withPride = loadFont(
    'inclusive/Gilbert_TypeWithPride.otf'
  );
  whatisQueer = loadJSON(
    'inclusive/voices.json'
  );
}
```

Performer #2:

'*If program code is like speech inasmuch as it does what it says, then it can also be said to be like poetry inasmuch as it involves both written and spoken forms.*'
(Cox, 2013, p.17)

'*Speech grounds language in the voice, the orientational metaphor grounds semantics in the body. It follows that computer software cannot have access to systems of meaning without at least some kind of reference to bodily relationships [...] Programmers bring bodily meaning to their work by applying models of human perception, and by trying to account for the ways that other social bodies are drawn into the process of meaning production.*'
(Cox, 2013, p.26)

Performer #1:
· Make a deliberate pause for non-pronounced ( ) , { } ;

```
function setup() {
  createCanvas(
    windowWidth,
    windowHeight
  );
  background(2.34387);
  makeVisible();
}
```

Performer #2:
· Pronounce all punctuation

There is more to coding than simply the demonstration of formal logic, as if everything could be reduced to input and output. Of course computers don't really speak but follow prescribed rules of execution, tasks, and actions. But, nevertheless, code can be broadly considered speech as it does what it says, and moreover does what it says at the moment of saying it. Any simple opposition of human and machines would be an oversimplification. Humans are not preprogrammed to execute their preprogrammed instructions and scripts, or 'input-output machines' as Dominique Laporte suggests in *A History of Shit* (2002).

Performer #2:
· Pronounce all punctuation except , “ ” { } ;

```
function SpeakingCode(
  iam,
  makingStatements
) {
  let getVoice = “inclusive/voices/”
    + iam + makingStatements + “.wav”;
  speak = loadSound(
    getVoice,
    speakingNow
  );
}

function speakingNow() {
  speak.play();
}
```

Performer #1:

In mathematics, zero is an important number and not to be dismissed as nothing. If we add a zero to the right side of any number, it is multiplied by ten. In Indian mathematics, the zero symbol counts for absence as well as space making it a much more positive sense of absence (Barlow, 2001, p. 35). Whereas, Leibnitz (working more in the Hebrew tradition of taking the void as the state from which the world was created) suggests the spirit of God belongs to the 'all-powerful One.' (quoted in Barlow, 2001, p.42)

Performer #2:

· *Omit { } , ; ( )*

· *Pronounce [ ] . =*

· *Make a deliberate pause for non-pronounced symbols*

```
function notNew(getQueer) {
  this.size = floor(
    random(
      15.34387,
      30.34387
    )
  );
  this.xxxxx = width/2.0;
  this.yyyyy = random(
    height / 3.0,
    height + 20.0
  );
  this.speed = random(
    2.34387,
    3.34387
  );
  this.gradient = 240.0;
}
```

Performer #1:

As Alain Badiou has it: 'we live in the era of number's despotism [...] Number governs our conception of the political […]', numbers govern science, history, cultural representations, the economy, our souls, 'But we don't know what number is, so we don't know what we are.' (2008, pp.1-4)

Performer #2:
· Pronounce everything

```
this.moveUP = function() {
  this.yyyyy += -this.speed;
  this.speed += sin(
    radians(
      (frameCount % 360.0) * this.speed
    )
  ) - 0.009;
};
```

Performer #2:
(together with Performer #1 on underlined parts)

'Whether *[…] gathering information, telecommunicating, running washing machines, doing sums, or making videos, all digital computers translate information into the zeros and ones of machine code. These binary digits are known as bits and strung together in bytes of eight. The zeros and ones of machine code seem to offer themselves as perfect symbols of the orders of Western reality, the ancient logical codes which make the difference between on and off, right and left, light and dark, form and matter, mind and body, white and black, good and evil, right and wrong, life and death, something and nothing, this and that, here and there, inside and out, active and passive, true and false, yes and no, sanity and madness, health and sickness, up and down, sense and nonsense, west and east, north and south. And they made a lovely couple when it came to sex. Man and woman, male and female, masculine and feminine: one and zero looked just right, made for each other: 1, the definite, upright line; the 0, the diagram of nothing at all: penis and vagina, thing and hole… hand in glove. A perfect match.*'
(Plant, 1997, pp.34-35)

Performer #1:
(together with Performer #2 on underlined parts)
· Omit { } " " ; ( )
· Pronounce . = <= ||
· Make a deliberate pause for non-pronounced symbols

```
this.isInvisible = function() {
  var status;
  if (
    this.yyyyy <= 4.34387 ||
    this.yyyyy >= height + 10.34387
  ) {
      status = "notFalse";
    } else {
      status = "notTrue";
    }
  return status;
};
```

Performer #2:

Although it takes two to make a binary (and set up the heterosexist paradigm), clearly inequalities are expressed in the tendency to privilege one side of the equation over the other - with positive and negative attributes accordingly.

Performer #1:
· Pronounce everything

```
this.shows = function() {
  textFont(withPride);
  textSize(this.size);
  textAlign(CENTER);
  this.gradient -= 0.5;
  noStroke();
  fill(this.gradient);
  text(
    getQueer,
    this.xxxxx,
    this.yyyyy
  );
};
```

Performer #2:

'*C+=, the world's first truly feminist computer programming language. Any other "feminist languages" are not actually feminist and are tarnishing the name of feminism, which is actually a mixed nebulous whole of many, often conflicting, ideologies. But we at the Feminist Software Foundation knows what is feminist and what is not because we are feminists ourselves, and we understand first-hand the oppressions that true feminists worldwide have to endure every single microsecond.*' (Feminist Software Foundation, 2013)

And from the *C+= manifesto* (Feminist Software Foundation, 2016):

'*Booleans are banned for imposing a binary view of true and false. C+= operates paralogically and transcends the trappings of Patriarchal binary logic. No means no, and yes could mean no as well. Stop raping women.*'

'*Instead of Booleans we now have Boolean+, or bool+ for short, which has three states: true, false, and maybe. The number of states may go up as intersectionality of the moment calls for such a need. […] No class hierarchy or other stigmata of OOP (objectification-oriented programming). In fact, as an intersectional acknowledgement of Class Struggle our language will have no classes at all.*'

Performer #1:
· *Omit ( ) { } ; [ ] " "*
· *Pronounce = <= == ++ . % , -*

```
function draw() {
  background(2.34387);
  for (
    let non_binary = floor(0.34387);
    non_binary <= queerRights.length
      - floor(1.34387);
    non_binary++
  ) {
      queerRights[non_binary].moveUP();
      queerRights[non_binary].shows();
      let status = queerRights[non_binary]
        .isInvisible();
    if (status == "notFalse") {
      queerRights.splice(
        non_binary,
        floor(1.34387)
      );
    }
  }

  if (
    (queerRights.length <= 2.0) &&
    (frameCount % 20 == 4.0)
  ) {
    makeVisible();
  }
}
```

Performer #1:

Alan Turing uncracked codes that others couldn't understand but that served to endorse the idea that he was also a cracked code in himself, eventually found guilt of 'gross indecency' in 1952. And the historical facts collapse into allegory. First of all, he was proscribed oestrogen to reduce his sexual urge, under the dubious logic that to all intensive purposes he was female - this was a reversal of earlier judgements to give gay men testosterone to make them more male, yet ironically making them sex machines. (See Andrew Hodges's *Alan Turing: The Enigma*.)

Sadie Plant concludes the Turing story: 'Two years later he was dead [...] By the side of the table was an apple, out of which several bites had been taken.' And this queer tale does not end here. There are rainbow logos with Turing's missing bytes on every Apple Macintosh machine.' (1998, p.102)

He loved Snow White.

Performer #2:

But we seem to have come a long way since the claims and counter claims of A.I.: in proving yourself to be 'human', 'not human' or 'not not human'.

The so-called 'post-humanities' develops this challenge to move beyond established forms and methods of disciplinary knowledge. For Rosi Braidotti, the idea of the 'human' is enmeshed in the larger anthropocentric problems that considers traditional humanism as no longer able to fully account for the human's entangled, complex relations with animals, machines, the environment, and planetary computation (2013). The humanities needs an upgrade to include the 'more-than-human condition'; actor-network theory, feminist new materialisms, environmental humanities, systems theory, software studies, science and technology studies, human-animal studies, trans, queer, anti-imperialist theory-practices, and other post- or non-disciplinary studies.

For Braidotti, the humanities has a lot to answer for, and ethics needs to be expanded beyond the frame of (White, Western, heterosexual) 'man' as the signifier of all rationality and reason. The universalist ideology associated with humanism is inherently far too narrow and flawed - if not fascist in tone.

Now everything is thoroughly 'entangled'.

Performer #1:

Perhaps what is at stake is a deeper way into what Karen Barad would call 'entanglements' of matter and meaning (2007).

She is referring to both the 'uncertainty principle' that confirms the trade-off between knowing more or less about position and momentum, and to Niels Bohr's 'complementarity principle' as a means to understand how individual things have their own independent sets of determinate properties and yet other properties remain excluded (2007, p.19). Her point is that causes and effects work through intra-actions, and these operate through determinate phenomena and exclusions, and hence are always open-ended: indeterminacy, contingency and ambiguity coexist with causality and determinacy.

Performer #2:
· Omit ( ) { } ; [ ] " " +
· Pronounce = / .

```
function SpeakingCode(
  iam,
  makingStatements
) {
  let getVoice = "inclusive/voices/"
    + iam + makingStatements + ".wav";
  speak = loadSound(
    getVoice,
    speakingNow
  );
}

function speakingNow() {
  speak.play();
}
```

Performer #1:
(together with Performer #2 on underlined parts)

Queer is... making binaries strange.

'If "queer is" is answered in the interface version of the piece, it is not so much the given suggestions in the meaning of the sentences, the content, which are the answers. "Queer is..." becomes the collective of voices, the disorder, which escapes its attribution to any speaking subject and any representations; "queer" becomes pure expressions. Queer, as the in-between, replaced, deterritorialized, is, hence, captured not in the content, but in the non-structure in the form of the interface version, described in a uniformity in the source code, compiled into a machine code and delivered by an illegible moment of execution, the causal interpretation.' (Muldtofte, forthcoming)

Performer #2:
(together with Performer #1 on underlined parts)
· Omit ( ) { } ;
· Pronounce = == . ,

```
function makeVisible() {

  queers = whatisQueer.queers;
  let addQueers = floor(
    random(2.34387, 4.34387)
  );
  let makingStatements;

  for (
    let gender = floor(0.34387);
    gender <= addQueers;
    gender++
  ) {
    let WhoIsQueer = floor(
      random(queers.length)
    );
    if (
      queers[WhoIsQueer].statement3 == "null"
    ) {
      queerRights.push(new notNew(
          queers[WhoIsQueer].statement2
        ));
      makingStatements = 2.0;
    } else {
      makingStatements = floor(
        random(2.34387, 3.34387)
      );
      if (makingStatements == abs(2)) {
        queerRights.push(new notNew(
            queers[WhoIsQueer].statement2
          ));
      } else {
        queerRights.push(new notNew(
            queers[WhoIsQueer].statement3
          ));
      }
    }

    if (gender == abs(2)) {
      SpeakingCode(
        queers[WhoIsQueer].iam,
        makingStatements
      );
    }
  }
}
```

Performer #1 + Performer #2:

Queer is... making binaries strange.

Bibliography

Badiou, A. (2008) *Number + Numbers*. Cambridge: Polity.

Barad, K. (2007) *Meeting the Universe Halfway: Quantum Physics and The Entanglement of Matter and Meaning*. Durham: Duke University Press.

Barlow, JD. (2001) *The Book of Nothing*. London: Vintage.

Braidotti, R. (2013) *The Posthuman*. Cambridge: Polity.

Cox, G. (2013) *Speaking Code: Coding as Aesthetic and Political Expression*. Cambridge, Mass: MIT Press.

Feminist Software Foundation. (2013) *Feminist Software Foundation: C-Plus-Equality.* [online] Available at: https://github.com/ErisBlastar/cplusequality/blob/master/hellofeminists.Xe [Accessed 13 Apr. 2018].

Feminist Software Foundation. (2016) *Feminist Software Foundation: C-Plus-Equality.* [online] Available at: https://github.com/ErisBlastar/cplusequality/blob/master/README.md [Accessed 13 Apr. 2018].

Hodges, A. (1983) *Alan Turing: The Enigma*. London: Walker Books.

Laporte, D. (2002) *A History of Shit*. Cambridge, Mass., London: MIT Press.

Muldtofte, L. (forthcoming) *Language Plus Code,* PhD thesis. Aarhus University.

Plant, S. (1998) *Zeros + Ones: Digital Women and the New Technoculture*. London: Forth Estate.

Colophon

Vocable Code (13082018)
A lecture-performance in six parts
by Winnie Soon & Geoff Cox

Published on ‡ DobbeltDagger 2018
ISBN 978-87-970443-1-5
https://dobbeltdagger.net
Web + print by Anders Visti

Project info: http://siusoon.net/vocable-code/